AFRL-IF-RS-TR-2005-12
Final Technical Report
January 2005

HUMINT EXTRACTION AND FUSION SYSTEM

General Dynamics

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-12 has been reviewed and is approved for publication

APPROVED: /s/
SHARON M. WALTER
Project Engineer

FOR THE DIRECTOR: /s/
JOSEPH CAMERA, Chief
Information & Intelligence Exploitation Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 2005	**FINAL** **Aug 02 – Sep 04**

4. TITLE AND SUBTITLE

HUMINT EXTRACTION AND FUSION SYSTEM

5. FUNDING NUMBERS

C - F30602-02-C-0174
PE - 63789F
PR - 407E
TA - HU
WU - MI

6. AUTHOR(S)

General Dynamics: Michael Schiller, John Gucwa, Christopher Crowner, Jeannette Neal
BBN Technologies: Michael Crystal, Ralph Weischedel

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

General Dynamics
4455 Genesee Street
Buffalo NY 14225

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/IFED
525 Brooks Road
Rome NY 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TR-2005-12

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Sharon M. Walter/IFED/(315) 330-7890 Sharon.Walter@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 Words)***

The objective of this effort was to develop the HUMINT Processing Subsystem (HPS) which extracts and exploits information from text documents to help the Intelligent Fusion System (IFS) identify, locate, and track targets in hide and in the open. The goal of HPS was to accurately extract information from HUMINT (Human Intelligence) about targets to assist in candidate target identification and intent determination. Document sources include, but are not limited to, text-based message traffic and other text document sources such as Integrated Intelligence Production Reports (IIPRs), United States Message Text Formats (USMTFs), annotated imagery and imagery support data. Information of interest might include target name, size, location, movement, appearance, disappearance, etc. HUMINT typically includes both structured text (e.g., tables, lists) and free-form prose text (e.g., sentences, paragraphs), which can be processed to produce information of value to the IFS. Existing tools were tailored and extended to produce the HPS. The integration of the HPS with the IOTA (Infrastructure Operations Tool Access) system and GIP (Generic Intelligence Processor), known as the Information Extraction Processing System (IEPS), worked sufficiently at JEFX-04 to be recommended for transition.

14. SUBJECT TERMS

information extraction, information exploitation, intelligence fusion, text processing, JEFX-04

15. NUMBER OF PAGES 53

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

1.1 Background

The problem of finding, fixing, tracking, targeting, engaging and assessing (F2T2EA) stationary and moving targets is the subject of an applied research and development being executed by four partnering AFRL Directorates, principally under a Sensors Directorate (AFRL/SN) program entitled "Targets Under Trees (TUT) Family of Systems and Supporting Technologies" and an Information Directorate (AFRL/IF) program entitled "Intelligence Fusion for Targets-Under-Trees."

The AFRL/SN program is concerned with the overall TUT systems engineering, development for the Foliage-Penetrating (FOPEN) radar, its associated technologies and other support functions, including modeling and simulation. The weapon fuse technology development is being conducted through the Munitions Directorate (AFRL/MN); and the human-system interface technology development and design is being conducted by the Human Effectiveness Directorate (AFRL/HE).

The AFRL/IF Intelligence Fusion for Targets-Under-Trees program addresses Intelligence Fusion and its supporting technologies. The AFRL/IF TUT program (hereafter referred to as the Intelligence Fusion System, TUT IFS, or IFS) is chartered to develop a capability for performing the Find, Fix and Engage portions of the kill chain process on targets employing concealment with camouflage and foliage. The IFS concept is to find and identify concealed mobile ground vehicles using multi-sensor fusion and Very High Frequency (VHF) Synthetic Aperture Radar (SAR) employing Change Detection (CD) techniques. The developments required to meet the IFS objectives involve FOPEN radar with Change Detection, information fusion, terrain characterization, weapons fuze and human-system interface technologies.

The Intelligence Fusion System was developed to address the combatant commanders concerns to effectively, identify, fix and target adversary resources and assets employing camouflage, concealment and deception. Although the IFS is applicable to strategic, operational and tactical levels of conflict, the emphasis to date has been on the tactical level. It is at this conflict level where combatant commanders have expressed their greatest concern in rapidly identifying, fixing and targeting critical assets that may have the greatest impact on successful mission execution.

A major design goal of the IFS is to process, fuse and display the following types of information as a minimum: Ground Moving Target Indicator (GMTI), Signal Intelligence (SIGINT) and Imagery Intelligence (IMINT). Operationally, this data would be made available primarily from the following collection platforms: Joint STARS, U-2, Global Hawk, Rivet Joint, Guard Rail, NTM and FOPEN radar systems.

A critical part of the IFS program is the development of supporting technology for the IFS, particularly multi-intelligence (Multi-INT) fusion technologies to enhance the ability to provide continuous track, location and identification of ground moving targets through fusion. This includes the potential integration of other intelligence data, for example, Video, EO/IR,

MSI/HSI, MASINT (seismic, acoustic, magnetic), HUMINT (HUMan INTelligence), and/or data fusion algorithms and capabilities.

Information Extraction (IE) technology attempts to facilitate the integration of intelligence data by identifying and extracting important information from unstructured text documents and messages. IE technology has been the subject of numerous research, development and Advanced Technology Demonstration (ATD) programs sponsored by the Air Force Research Laboratory's Information Directorate (AFRL/IF), including: the Document Content Analysis and Retrieval System (DCARS), the Intelligence Analyst Associate (IAA), the Text Exploitation ATD (TEA), IAA-CYC, the Intermediate Text Exploitation ATD (ITEA), and the Automated Information Extraction Systems (AIES) programs (the latter is a cooperative program with the Joint Warfare Analysis Center (JWAC)). IE can benefit the IFS in the near-term by extracting target-related information from HUMINT such as target names and associated locations and times. This capability will augment the information acquired by the IFS from SIGINT sources to help increase the precision and completeness of target identification and tracking.

The objective of the research described in this report was to develop the HUMINT Processing Subsystem (HPS) which extracts and exploits information from text documents to help the Intelligent Fusion System identify, locate, and track targets in hide and in the open.

1.2 HUMINT Processing Subsystem (HPS) Program Goals

The goal of HUMINT Processing Subsystem (HPS) was to accurately extract information from HUMINT about targets to assist in candidate target identification and intent determination. The HPS extracts and processes relevant information from text-based documents/messages, including structured text portions such as tables and the free-form prose text of those documents. Information of interest might include target name, size, location, movement, appearance, disappearance, etc. In the context of this effort, HUMINT can more generally be defined as text-based message traffic and other text document sources composed by humans, particularly the free-form prose text of these messages/documents that can be processed to produce information of value to IFS. Document sources include, but are not limited to, text-based message traffic and other text document sources such as Integrated Intelligence Production Reports (IIPRs), United States Message Text Formats (USMTFs), annotated imagery and imagery support data. The output of the HPS is not HUMINT, but is rather information extracted from HUMINT that indicates what targets were present (or missing), where, and when. Existing tools were tailored and extended to produce the HPS.

The HPS was developed such that it can be integrated into the IFS as a service available to the IFS Fusion Manager through the eXtensible Distributed Architecture (XDA) "backbone" of the IFS. The HPS was developed to be compliant with the requirements needed for integration with the Target Under Trees (TUT) IFS and subsequent delivery as part of the IFS to the Distributed Common Ground Station (DCGS) and/or the Intelligence, Surveillance and Reconnaissance (ISR) division of the Air Operations Center (AOC). HPS was also enhanced and adapted for participation in the Joint Expeditionary Force Experiment that took place in Summer 2004 (JEFX-04).

The HPS Program was sponsored and acquired by the Air Force Research Laboratory (AFRL) Rome Research Site (formerly, Rome Laboratory). This effort was performed by General Dynamics Advanced Information Systems (GDAIS; previously, Veridian Engineering) as the prime contractor with BBN Technologies as subcontractor.

2 Program and Software Development Overview

2.1 HPS Program Overview and Milestones

This HPS Program consisted of two parts:

1. Phase 1 - HPS Development. An 18-month effort consisting of three development spirals during the performance period of August 2002 through February 2004.
2. Phase 2 - HPS Customization & Support for JEFX-04. A 7-month effort consisting of three JEFX-04 spirals the performance period of February 2004 through September 2004.

During the Phase 1 development part, the HPS system was incrementally developed through a sequence of increasingly more full-functioned operational prototypes, each developed and delivered to AFRL during one of the three successive development spirals. The final operational HPS prototype resulting from the HPS Development phase was delivered to the Government in February 2004.

The main Phase 1 development phase of the HPS program consisted of three spiral development periods during the 18 month performance period. The first table below lists the Technical Interchange Meetings (TIMs) that were held at AFRL Rome Research Site during Phase 1. In addition to our HPS team, the TIMs frequently included the participation of representatives of other contractors working on the IFS program. Such contractors included Orincon (prime contractor on the IFS program) and BAH (Booz Allen Hamilton), for example.

Table 1: HPS Technical Interchange Meetings

Phase 1. HPS Development Program (Aug 2002 – Feb 2004)		
HPS Spiral	**Purpose**	**TIM Date**
0	Kickoff	7 October 2002
		14 January 2003
		26 February 2003
1		16 June 2003
	Integration	4 September 2003
		19 February 2004
2		16 October 2004
3	Final TIM	19 February 2004

During HPS program Phase 1, each spiral included at least one delivery and installation of the HPS software system at AFRL Rome. Table 2 lists the sequence of increasingly more fully functioned HPS system versions that were delivered to the Government through Phase 1 and 2. Each system version in the sequence was equipped with more capabilities compared to the previous version and was closer to meeting the targeted user requirements for the program. The table also provides the date of each of the deliveries.

Table 2: HPS Software Deliveries

Phase 1. HPS Development Phase (Aug 2002 – Feb 2004)		
Spiral	**HPS Version**	**Delivery Date**
1	1.0	5 June 2003
2	1.1	23 September 2003
3	1.2	21 January 2004
Final	1.3	18 February 2004
Phase 2. HPS Enhancement for JEFX-04 (Feb 2004 – Aug 2004)		
Spiral	**HPS Version**	**Delivery Date**
2	1.3.1	26 March 2004
3	1.3.2	27 April 2004
3	1.3.3	13 May 2004
3	1.3.3a	22 June 2004
Main Exercise	1.3.3.1	15 July 2004

During HPS program Phase 2, GDAIS supported the Government with HPS enhancements and participation in the JEFX-04 three spirals and main event. GDAIS tested the HPS on JEFX-04 data, made enhancements to the HPS software and knowledge resources to adapt the software and improve performance on the JEFX-04 task and domain, collaborated with the Government and other contractors to integrate the HPS into the IFS, and made enhancements to the software to improve performance, reliability, and robustness. Deliveries of the increasingly enhanced and tailored HPS system were made to the Government as a part of each spiral and for the JEFX-04 main event as shown in Table 2, above. Table 3 presents the dates of the periods during which GDAIS representatives were on-site at Nellis AFB to support and participate in the JEFX-04 activities.

Table 3: JEFX-04 On-Site Participation/Support for Spirals at Nellis AFB

Phase 2. HPS Enhancement for JEFX-04 (Feb 2004 – Aug 2004)		
JEFX Spiral		**On-Site Participation/Support Dates**
2		25-29 March 2004
3		12-21 May 2004
Main Exercise		15-23 July 2004

2.2 Software Development Process

Our team used a spiral incremental software engineering process that iterated on the software development tasks. During each spiral, the team refined the definition of the users' problems, analyzed and prioritized requirements and selected the requirements to be addressed during that particular cycle, developed high and low level software designs for the capabilities addressing the selected requirements, implemented the capabilities in software code, performed several

types of testing on the code and prototype system, installed the software prototype at AFRL Rome for their evaluation, provided familiarization training to AFRL personnel on how to use the system and its new functionality, and gathered evaluation feedback from users. The evaluation feedback was used as input to the next development spiral to guide the development of the next version of the system so as to deliver a more fully functional prototype to better fulfill user needs. The feedback potentially affected all steps of the development process, but most importantly was used to prioritize and select requirements for the next prototype development period. Figure 1 illustrates the overall process and the spiral nature of the process.

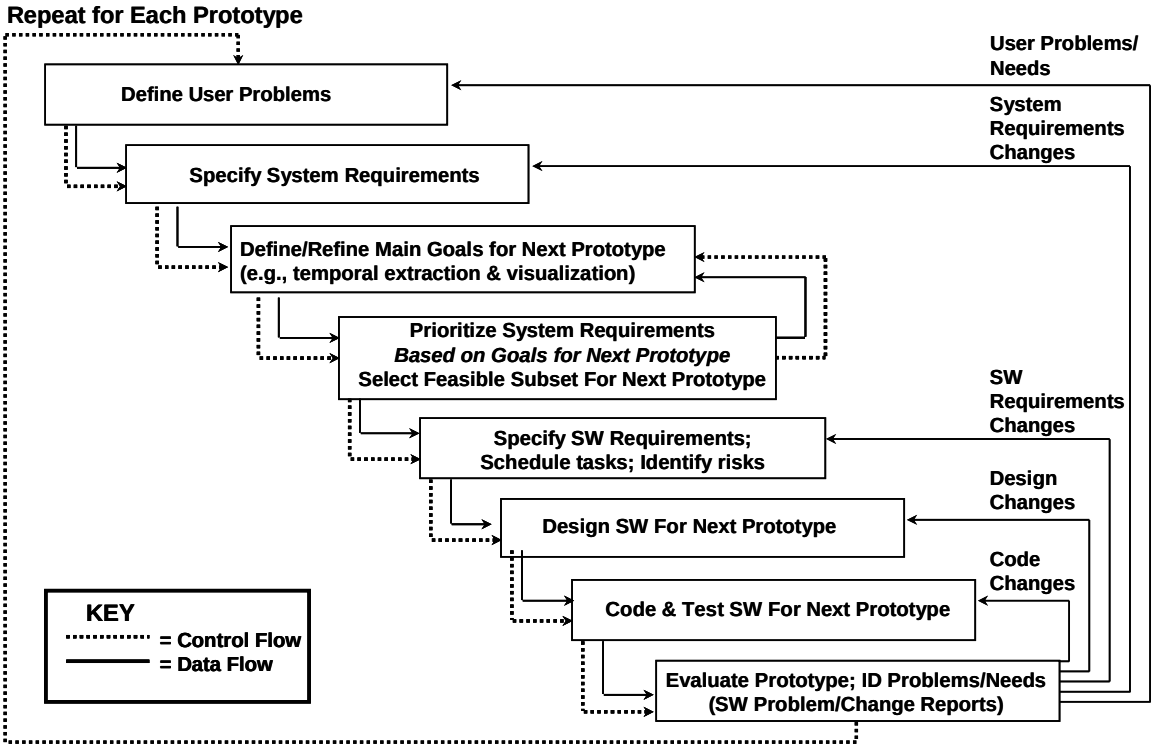

Figure 1: HPS Spiral Software Development Process

Evaluation feedback from the Government was gathered as part of each spiral increment and was used to update the Concept of Employment, Requirements Specification, design, and other documents and specifications generated during the software development process. User input (feedback) was used to guide the software development during the next development cycle. It was used to prioritize and select requirements for the next system incremental spiral development period.

An important feature of this model is that it provides users and other Government personnel with visibility into the development process, gives users the opportunity to periodically exercise and evaluate the software being developed, and enables them to guide the development to better meet their needs. This process model accommodates the changing user requirements that commonly

occur in the development process. The model is designed to lower risk and help ensure that the software development process culminates in delivery of a useful software system targeted to meet user needs.

Our team's approach to creating and refining system/software development documents implemented the concept of "working documents" that are developed incrementally, similar to the development of the software. As part of this approach, the documents were revised and updated through the HPS Program when appropriate. This approach supports better handling of dynamic emerging or changing user problems and requirements. So, for example, a change in a user problem and associated requirements is reflected in a modification to the Concept of Employment and the corresponding requirements in the Requirements Document. In addition, the ramifications of these additions/changes to the problem definition and user requirements are then carried through as modifications to the design documentation, and other relevant documents.

3 System Overview

The HPS is integrated into the IFS as a service available to the IFS Fusion Manger through XDA. Figure 2 below shows the IFS high level architecture with the HPS plugged in via XDA.

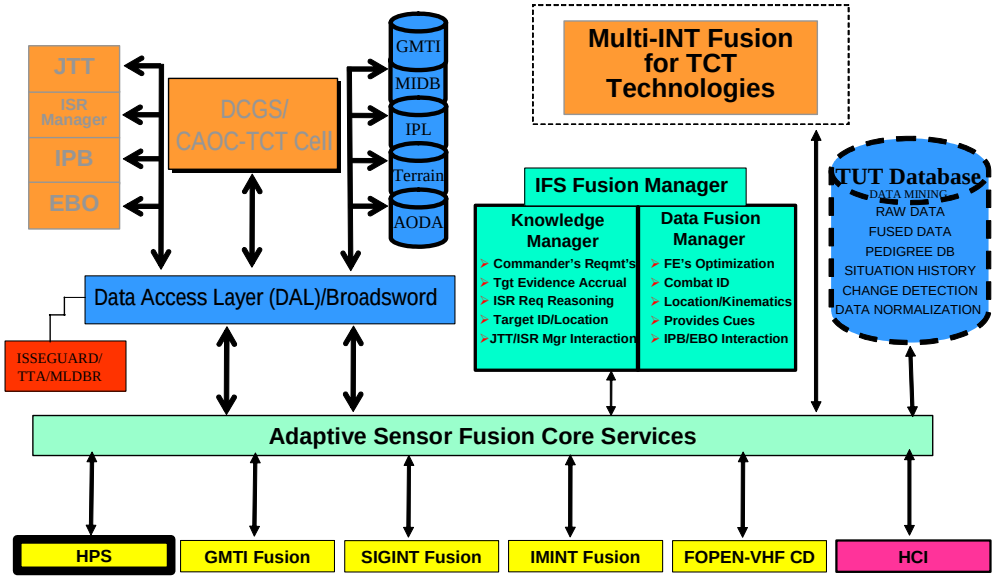

Figure 2: The HPS is a Service Available to the IFS via XDA

The following is a high level summary of what the HPS is designed to extract and output:

- *Facilities and Equipment.* While Equipment entities are the primary targets of interest, Facilities are often used as reference points for locating equipment in HUMINT. Facilities are therefore extracted and made available to the IFS. HPS extracts Facilities and Equipment at two levels of specificity: Class and Instance. Most often in HUMINT the text refers to a member of a class of objects, for example, "two T2 tanks", "the pickup truck." A class reference is less precise than an instance reference like "the T2 tank with serial number 636." Extraction of a Class reference provides target type, but not target instance.
- *Persons and Organizations.* Again, while Equipment entities are the primary targets of interest, Persons and Organizations, beyond being of interest in general, can often be indirect references to equipment in HUMINT. Person and Organization references are therefore extracted and made available to the IFS. For example, a military unit (e.g., brigade) will be identified as an organization. This is in fact an indirect reference to all of the equipment within that unit. In the future, real world knowledge could be used by the system to decompose a unit in to its components for a more automated extraction process.
- *Associated Date-Time.* Information regarding the date and/or time that a target was reported to be at a location was to be extracted. Dates/times were normalized to a Date/Time Group (DTG) and reported date-times included an associated error interval (margin for error).
- *Associated Location.* Locations of targets are normalized to a latitude and longitude, and reported locations include an associated error extent (margin for error).
- *Associated Characteristics.* Potentially valuable target characteristics are available in HUMINT. For example, count (the number reported at a location and time), state (e.g., present, missing, damaged, etc), size, color, direction of travel and speed. State (at least

present or missing) seems directly related to Change Detection. Some of the other characteristics could provide target identification evidence to be accrued.

In addition to the above described output, a confidence measure can be associated with each identified entity (i.e., the likelihood that the text referred to the real world entity (instance) or type (class) indicated by the normalized form), associations between equipment and facilities and their characteristics, times and locations (i.e., the likelihood the equipment or facility is validly associated with the characteristic, time, or location in the real world). A single numeric confidence is difficult to achieve at this point in the development. Confidence measures were reported for extraction items based on testing (e.g., "high", "medium", "low"), reporting source (from the text: "new", "reliable", etc.), sighting (from the text: "possible", "probable", "confirmed").

The HPS has been built using an existing common architecture as the framework. The framework includes capabilities for control of, and communication between, the components and provides a plug-in, plug-out capability. Figure 3 depicts the high level architecture for the HPS.

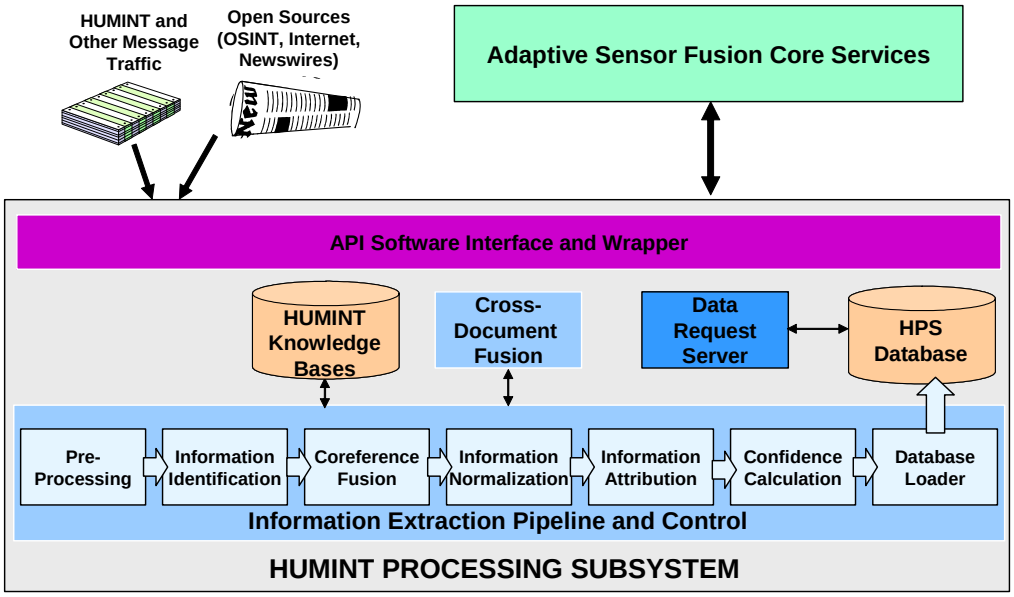

Figure 3: Architecture for HUMINT Processing System (HPS)

The high-level components/capabilities for the HPS are listed in Table 4. Implementations of some of these capabilities were provided by existing information extraction (IE) components, some more fully developed than others. Where possible, existing components have been enhanced and tailored to fulfill the HPS requirements. Components have been selected for enhancement and incorporation into the HPS based on the degree to which they meet the software requirements for the HPS capabilities and the degree to which they could easily be tailored and enhanced. For capabilities for which there are no completely satisfactory software components, satisfactory software components continue to be sought through surveys and study of the current and emerging IE technology.

Table 4: HPS Components/Capabilities

COMPONENT	PURPOSE
Information Extraction Pipeline and Control	Integrate the steps in the information extraction process and handle the control of each extraction component and the data transfer between components.
Preprocessing	Text zoning to determine the structure and parts of a message / document; identify extraneous text such as page headers and footers. Sentence breaking to break up the text of a document into individual sentences for further analysis. Text type determination to segment the text into strictly formatted, prose text and structured prose portions.
Information Identification	Find and identify text segments that express/mention relevant targets, locations, temporal information, and target characteristics in the messages or documents.
Coreference Fusion (Within-Document)	Resolve which text expressions within a document refer to the same entities, namely targets, locations, and dates/times. Perform discourse context tracking within document (to carry along a representation of the topic; e.g., the person, place, time, target, etc. that are assumed though not necessarily explicitly stated in clauses of the document).
Information Normalization	Generate a standardized form for each information item, namely targets, locations, dates/times, and characteristics.
Information Attribution	Assign (attribute) the identified locative, temporal, and characteristic information to the correct target(s) mentioned in the text, thus creating a relationship between the mentioned target(s) and their location, time, and characteristics.
Confidence Calculation	Calculate a credibility measure for each information item. Combine measures when information items are fused/merged.
Database Loader	Load the extracted information into the database.
HUMINT Knowledge Bases	Store and provide vocabulary terms and real world information on targets and locations of interest.
Cross-Document Fusion	Resolve which text expressions occurring in different documents refer to the same entities, especially targets and locations. Merge information extracted from different documents when appropriate for these entities.
HPS Database	Stores the extracted and merged/fused information. Provides the source of information with which to respond to requests from the IFS via the HPS API.
Data Request Server	Service data requests for extracted information. Includes requests for currently stored information and requests for information to be extracted in the future (alerts).
API Software Interface and Wrapper	Enable the HPS to be integrated into the IFS as a service “plugged into” the IFS via XDA. Provides the IFS with access to the services of the HPS.

4 Concept of Employment

The following subsections present applications of the information extracted from textual data, or, in other words, concepts of employment. These are candidates as to how the HPS may be employed in the context of IFS to support the goals of the IFS program. Information extraction can be applied in each case, but the focus of the extraction may be different and the formats of the data within the text may be different if different sources apply to different problems.

4.1 Situational Awareness

IFS benefits from a priori knowledge in the form of situational awareness, particularly during the Fix and Assess Steps described in the Concept of Employment for the IFS. In particular the Reasoning Engine can employ backward chaining and can leverage the situation model to produce an identification of the target. The more current the situation model is, the more accurate (and more likely found) the identification will be. Using the data extracted from textual information to update or augment the situational model (as stored in an external database) used by IFS will improve identification. This integration is perhaps easier than integrating HPS data in real-time. The extraction effort would be focused on updating order of battle information by extracting target location information and Bomb Damage Assessments (BDAs), that is, "Where is the equipment stationed, has it been moved and is it still in play?" In a sense, this mode of operation employs the HPS to function as a mechanism for virtually updating order of battle resources such as the Modernized Integrated Database (MIDB). A simple example of this mode of operation is illustrated in Figure 4 and described below.

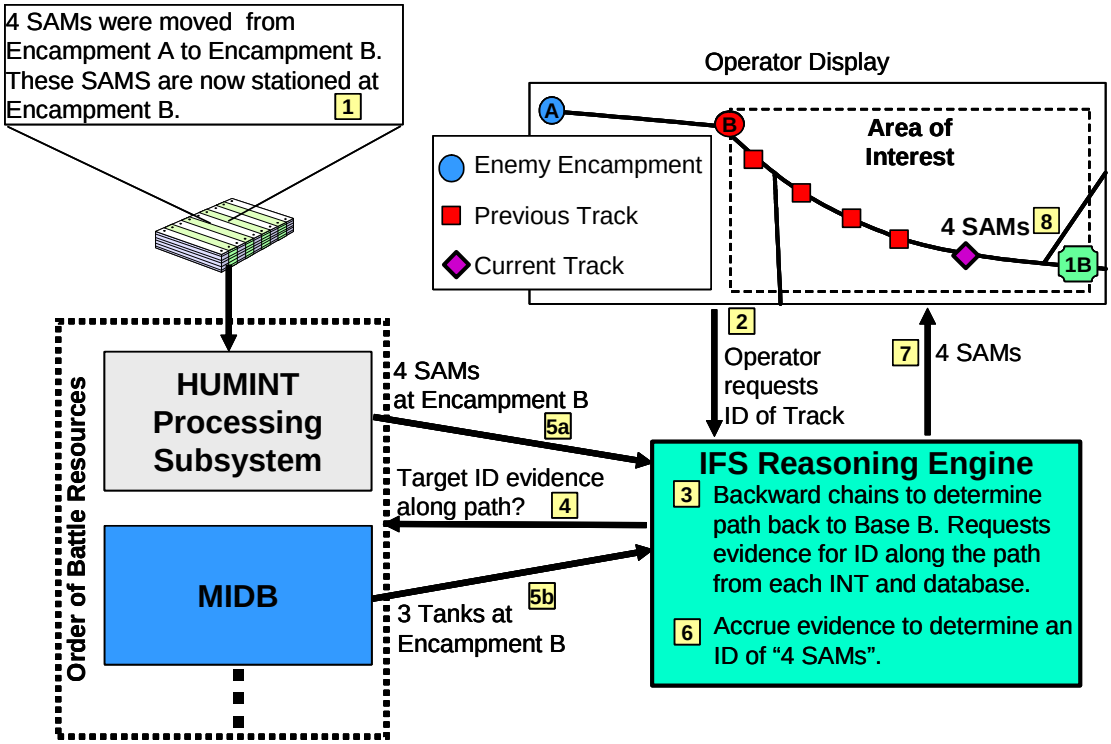

Figure 4 Situational Awareness Mode Example

- Step 1 (indicated by the 1 in the box in the figure): A message is received and processed by the HPS. The message indicates four SAMs were moved from Encampment A to Encampment B. After the HPS processes the message the following information is available in the HPS database:
  - 4 SAMs are missing from the location of Encampment A as of the time of the report.
  - 4 SAMs are located at the location of Encampment B as of the time of the report.
  - Additional information could be inferred by applying real world knowledge. For example, if the knowledge as to how fast SAM systems can move and the location of the road from Encampment A and B were available, estimates of locations and times along the road could be made. This is beyond the scope of this effort. The extracted information is based solely on the content of the messages.

  In this mode the HPS is treated as an Order of Battle Resource, so the information is not published. The HPS will provide data when queried by the Reasoning Manager.
- Step 2: A track, generated by MTI, enters an operator's area of interest. The operator clicks on the track and requests a target ID for the track. There is no ID available in the pedigree for the track. The operator indicated ascertaining an ID is critical, so the request is forwarded to the IFS Reasoning Engine.
- Step 3: The IFS Reasoning Engine backward chains through the stored track information. Evidence is accrued from the INTs and Order of Battle Resources are queried for information on targets along the path.
- Step 4: Queries are passed to each of the Order of Battle Resources, including HPS and MIDB, for ID evidence along the path.
- Step 5: Each database reports the evidence it holds as to target ID. HPS (a) reports there are four SAMs at the location of Encampment B (a point along the backward chain). MIDB (b) reports there are three Tanks at the location of Encampment B. The HPS database does not contain the information that there are three Tanks at Encampment B since no message indicating their presence has been processed. The MIDB does not contain the information that there are four SAMs at Encampment B since it has yet to be updated with the new information contained in the message.
- Step 6: The IFS Reasoning Engine accrues evidence from all the INTs and databases. The evidence indicates the targets in the track are four SAMs to a high level of confidence.
- Step 7: The ID and confidence are reported to the operator's display.
- Step 8: The target ID is displayed for the operator.

4.2 Target Identification

The IFS would benefit from real-time information about targets that can be extracted from textual data. It is not reasonable to assume that there will be no latency in the availability of HUMINT (or other textual data), but certain information may be available in a timely enough fashion to be of value in updating target information for use by the Fusion Manager. The Reasoning Engine has more flexibility built in to accommodate latency as it accrues evidence and would also have the data from the HPS available to draw upon for evidence.

4.2.1 Backward Chaining

It would be valuable to have a capability to apply a method similar to backward chaining to associate HPS data with tracks during fusion. This mode would require the Fusion Manager to accommodate "late reporters", such as HUMINT and IMINT, and fuse the late reports with potential existing tracks (i.e., add the information from the late reports to the pedigree for potentially corresponding tracks). This differs from (and is simpler than) the Reasoning Engine's backward chaining in that it only deals with late reporters and existing tracks. The extraction effort would be focused on detecting target attributes that would be of value in merging tracks and improving identification, such as, "What is the target and what characteristics does it have?"

A simple example of this mode of operation is illustrated in the figure below and described in the following paragraphs.

- Step 1 (indicated by the 1 in the box in the figure): When the operator starts his shift, he defines an area of interest which is sent to the Fusion Manager.
- Step 2: The Fusion Manager subscribes to the INTs for the operator defined area of interest and begins fusing incoming information to produce tracks.
- Step 3: HPS (logically; this relationship is really handled by the XDA server) receives the subscription and records the area of interest.
- Step 4: After four track points (labeled 1 to 4) have been produced along the track by the Fusion Manager and shown on the Operator Display, a HUMINT message is received and processed by the HPS. This report refers to a sighting near a track point three points back from the current point (labeled 1). After the HPS processes the message the following information is available in the HPS database:
  - 2 vehicles with a characteristic of "3 meters long" were located at 491200N-264655E at 2320 along with associated confidences.
- Step 5: The extracted information is published and received by the Fusion Manager.
- Step 6: The Fusion Manager fuses the late report (extracted information outside of the normal track time window) with any applicable current tracks. The Fusion Manager also stores the information in the IFS XDA Historical Database for potential later use.
- Step 7: The Fusion Manager successfully fuses the HPS data with the current track and adds the new information to the track's pedigree.
- Step 8: When the operator requests pedigree information, the HPS data, "2 vehicles 3 meters long", along with associated confidences, is displayed along with the other pedigree information. Additionally, the source document for the information is available for drill down if the operator wishes.
- Step 9: After an additional track point (labeled 5) have been produced along the track, another HUMINT message is received and processed by the HPS. This report refers to a sighting near a track point three points back from the current point (labeled 2). After the HPS processes the message the following information is available in the HPS database:
  - 2 vehicles with a characteristic of "speed at least 30 K/h" were located at 491130N-264755E at 2323 along with associated confidences.

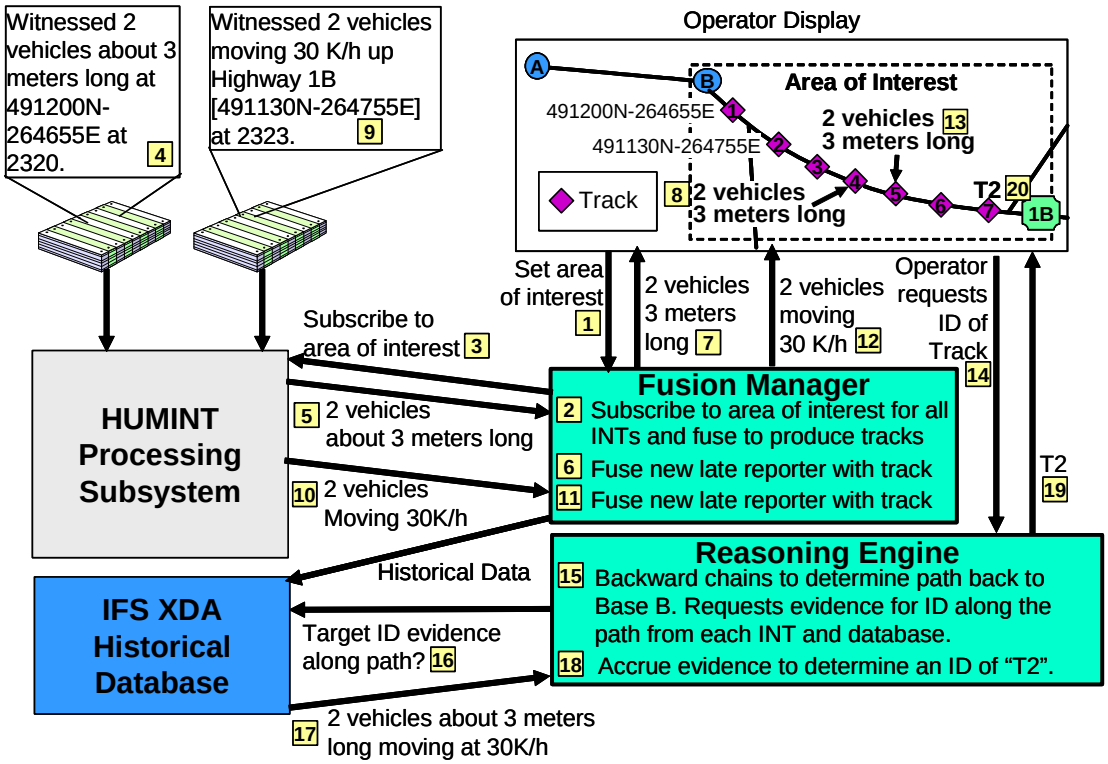

Figure 5 Target Identification Mode Backward Chaining Example

- Step 10: The extracted information is published and received by the Fusion Manager.
- Step 11: The Fusion Manager fuses the late report with any applicable current tracks and stores the information in the IFS XDA Historical Database.
- Step 12: The Fusion Manager successfully fuses the HPS data with the current track and adds the new information to the track's pedigree.
- Step 13: When the operator requests pedigree information, the HPS data, "speed at least 30 K/h" and "2 vehicles 3 meters long", along with associated confidences, is displayed along with the other pedigree information.
- Step 14: After two additional track points (labeled 6 and 7) have been produced along the track, the track is approaching an important intersection and identification becomes critical. The operator examines the pedigree and is unable to produce an exact ID, so requests an ID.
- Step 15: The IFS Reasoning Engine backward chains through the stored track information. Evidence is accrued from the INTs and Order of Battle Resources are queried for information on targets along the path.
- Step 16: The IFS XDA Historical Database is sent the query.
- Step 17: The HPS historical data, "speed at least 30 K/h" and "2 vehicles 3 meters long", along with associated confidences and information from other INTs is returned as a result of the query.
- Step 18: The IFS Reasoning Engine accrues evidence from all the INTs and databases. The evidence indicates the targets in the track are two T2 tanks to a high level of confidence.
- Step 19: The ID and associated confidence is reported to the operator's display.
- Step 20: The target ID is displayed on the operators display.

4.2.2 Forward Chaining

Applying forward chaining initiated by alerts from the HPS would be valuable. This mode would require the Fusion Manager to accommodate "late reporters", such as HUMINT and IMINT, and recognize information which will not be fused with a track but will rather issue an alert to the operator. The extraction effort would be focused on detecting changes in state, in particular from present to missing, that is, "Is a target still where it was last?"

A simple example of this mode of operation is illustrated in the figure below and described in the following paragraphs.

- Step 1 (indicated by the 1 in the yellow box in the figure): When the operator starts his shift, he defines an area of interest which is sent to the Fusion Manager.
- Step 2: The Fusion Manager subscribes to the INTs for the operator defined area of interest and begins fusing incoming information to produce tracks.
- Step 3: HPS (logically, this relationship is really handled by the XDA server) receives the subscription and records the area of interest.
- Step 4: After two track points (labeled 1 to 2) have been produced along the track by the Fusion Manager and shown on the Operator Display, a HUMINT message is received and processed by the HPS. This report refers to an activity at Base B. After the HPS processes the message the following information is available in the HPS database:
  - 1 mobile command post with a characteristic of "state missing" is located at 491215N-264625E at the time of the report along with associated confidences.
- Step 5: The extracted information is published by HPS and received by the Fusion Manager.

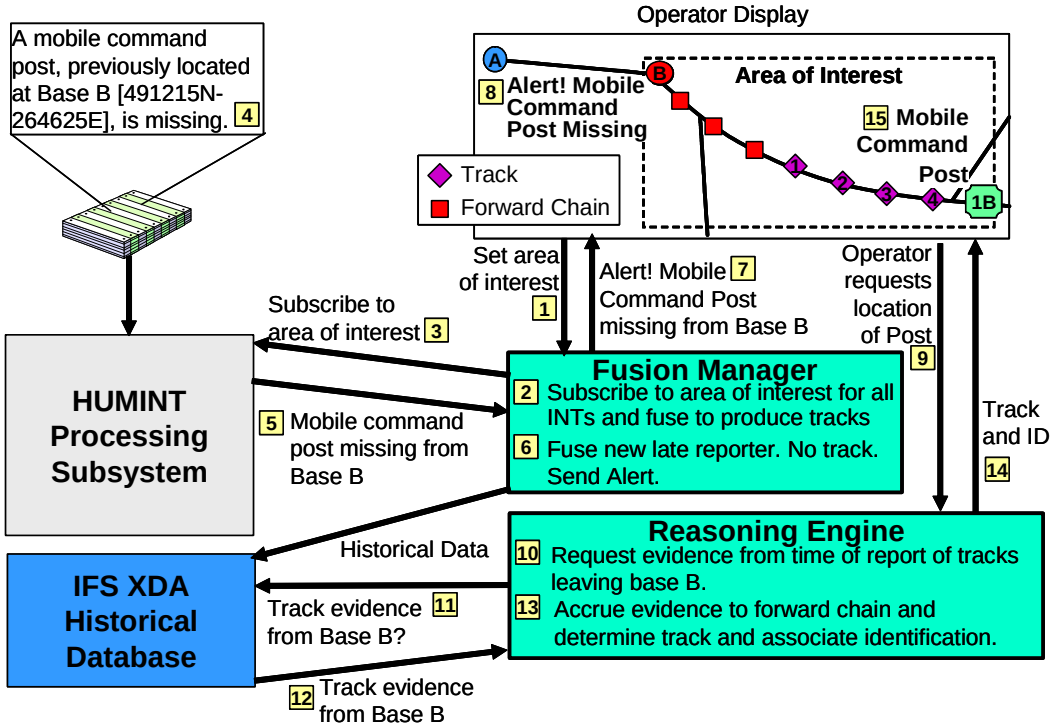

Figure 6 Target Identification Mode Forward Chaining Example

- Step 6: The Fusion Manager handles the late reporter and produces an alert. The Fusion Manager also stores the information in the IFS XDA Historical Database for potential later use.
- Step 7: The Fusion Manager sends the alert to the operator's display.
- Step 8: When the operator has configured the display to show alerts, the HPS data, "Mobile Command Post Missing" along with associated confidences, is displayed. Additionally, the source document for the information is available for drill down if the operator wishes.
- Step 9: The operator determines that the location of the command post is important and requests its location. Coincidently during this time, two additional track points (labeled 3 and 4) have been produced along the track.
- Step 10: The IFS Reasoning Engine requests historical data which provides evidence of targets leaving Base B.
- Step 11: The IFS XDA Historical Database is sent the query.
- Step 12: Historical Data from the INTs is returned as a result of the query.
- Step 13: The IFS Reasoning Engine forward chains (indicated by the boxes leading from Base B to the track) to determine the track and associate the ID.
- Step 14: The ID and associated confidence rating is reported to the operator's display.
- Step 15: The target ID is displayed on the operator's display.

4.3 Target Intent

A longer term goal of IFS is to ascertain the intent of targets. Textual data may be invaluable in ascertaining intent, however the types of information which can be extracted from textual data that indicate intent may be substantially different from those which indicate targets, target attributes, locations, and times. Substantial effort would be needed to define and implement extraction for intent. In all likelihood the requirements for intent determination would be different enough from those for situational awareness and target identification that this effort will be beyond the scope of this contract.

5 System Requirements

This section presents the high level system requirements for the HPS.

5.1 Functional

Requirements: The HPS shall:

1. Enable users of the IFS to use and exploit HUMINT and other text-based documents/messages in combination with the other INTs handled by the IFS.
2. Help users to identify, find, fix, and track targets of interest, especially mobile ground targets in the open and in hide.
3. Support the IFS capabilities for information fusion, geo-registration, and weapons fuze.
4. Support both the IFS backward chaining and forward chaining modes of operation to the extent that the latency of the HUMINT data allows.
5. Provide confidence measures for each of the extraction information items to indicate the likelihood that the information was extracted correctly and to indicate the likelihood that the extracted information reported by the HPS is true in the real world.
6. Process a variety of document types/sources including, but are not limited to, text-based message traffic and other text document sources, such as IIPRs, USMTFs, IDBTFs, GRAPHREPs, ATGs, BTGs, and annotated imagery and imagery support data.
7. Enable the system to be transitioned to user organizations such as the Distributed Common Ground Station (DCGS) and/or as applicable, the ISR division of the Air Operations Center (AOC).

5.2 Performance

Requirements: The HPS should:

1. Process text documents in near real time.
2. Respond to requests for information in near real time.

5.3 Software Quality

Requirements: The HPS shall have the following software quality factors:

1. Extensibility: the ease with which enhancement of the HPS can be accomplished.
2. Reusability: the degree to which the HPS can be used in other applications.
3. Usability: the ease with which input preparation, output access, and output interpretation for the HPS can be learned.
4. Flexibility: the ease with which users can manipulate and control various aspects of system processing to suit their individual needs and preferences.
5. Maintainability: the ease with which errors in the HPS can be located and corrected.
6. Security (integrity): the degree to which the HPS must control unauthorized access or modifications to system software and data.
7. Reliability: the degree to which the HPS must consistently perform its intended capabilities.
8. Interoperability: the degree to which the HPS must interface with other systems.

9. Correctness: the degree to which the HPS must satisfy its specified requirements.
10. Scalability: the degree to which the HPS scales up to handle large corpora of data.
11. Portability: the ease with which the HPS can be transferred from one hardware or software system environment to another.
12. Testability: the ease with which it can be ensured that the HPS performs its intended capabilities.

5.4 Software Interface

Requirements: The HPS shall:

1. Support integration into the IFS as a service available to the IFS Fusion Manager via XDA.
2. Provide access to all its services through the integration.

5.5 Design Constraints

Requirements: The HPS shall:

1. Be installable and executable on systems currently available in targeted test and operational sites, such as the AFRL Fusion Testbed, DCGS, and the ISR division of the Air Operations Center (AOC).

6 Technical Approach and Accomplishments

The following subsections provide an overview of our technical approach to the development of the HPS and the accomplishments achieved as part of this effort. This section focuses on the accomplishments of GDAIS. The accomplishments of subcontractor BBN are presented in Section 7.

6.1 Data Analysis

As part of requirements analysis and specification, GDAIS performed a data analysis task. The purpose of the data analysis task was to characterize the text types and document types in the HUMINT sample corpus provided by NASIC for the HPS development project and to specify requirements based on or derived from the nature of the data. In order to accurately obtain information from the documents in the sample corpus, it was necessary to classify the types of text in which that information can be found. As a result of the analysis, GDAIS categorized the documents into four types of text: Strictly Formatted Text, Structured Text, Free Text (Prose), and Unknown. Further, it was found that the presence and type of strictly formatted text found in any document depends largely on the classification of the document itself. Therefore, in addition to the analysis of the text types, it was necessary to examine the different types of documents in the sample corpus. The results of the investigation are documented in the document entitled "HUMINT Corpus Evaluation Working Document", dated 23 December 2002. NOTE: All numerical information reported in the following sections regarding the numbers of documents is approximate.

6.1.1 Summary of Analysis Results

An examination of the entire document set revealed the presence of five major types of documents:

- NIMA Imagery Interpretation Reports (USMTF Format)
- NIMA Imagery Interpretation Reports (Non-USMTF Format)
- CCJ2 Image Interpretation Reports
- Department of Defense Information Reports
- Central Intelligence Agency Information Reports

Other document types were also identified, but they did not comprise a large enough percentage of the documents to be worth further investigation at this time. These lesser-occurring document types include Inspection Reports, NIMA Intelligence Problem Cable Reports, Tactical Military Intelligence Summaries, and other types of Imagery Interpretation Reports.

A categorization scheme for the text within the documents was also developed. The four types of text used in the classification are:

(1) Strictly Formatted Text:
Text that adheres to some identifiable standard, such as the USMTF standard or a document-type-specific standard.

(2) Structured Text:
Text that appears in a structured, non-standardized format, often in the free text sections of documents. This includes tables that do not fall into the Strictly Formatted Text category and header information.

(3) Free Text (Prose):
Text that is comprised of typical English sentences and paragraphs.

(4) Unknown:
Any text that cannot be classified into one of the above specific categories.

The following table summarizes the Document Type information in the HUMINT Document Analysis.

Table 5 Data Analysis Results

Document Type	Frequency	Text Section Structure	Unique Attributes
Imagery Interpretation Report (NIMA v.1)	≈33%	- Sub-Header - Text - Image Data	- Adheres to the USMTF standard - Sub-Header includes location, subject information - Text is delimited with RMK/ or DES/ - All text lines begin with '/' - Free Text sections end with '//' - Image data is denoted by IMR/DTE:########
Imagery Interpretation Report (NIMA v.2)	≈7%	- Text, consisting of locations and descriptions	- Location line consists of a location description followed by a BE Number - Description is a free-text description of the location with any pertinent information that should be pointed out about that location
Imagery Interpretation Report (CCJ2)	≈4%	- Sub-Header - Text, consisting of itemized list of descriptions	- Large documents that summarize a large amount of data - Format of each item in list of images: Item ### with location information, followed by a free text description of the location, concluding with specific image data including location and time of the image
Department of Defense Information Report	≈20%	- Sub-Header - Text - Comments	- Mostly free-text documents - Sub-Header includes Country, Subject, Date, Source, and Summary Information - DOD Marquee is present that could be

			used to identify these documents.
Central Intelligence Agency Information Report	≈10%	- Sub-Header - Text - Comments	- Mostly free-text documents - Similar to the DOD Information reports, except with a CIA Marquee and a slightly different Sub-Header (which contains the same information)

Other distinct document types are also present in the sample corpus. These other types include Inspections Reports (≈0.3%), NIMA Intelligence Problem Cable Reports (≈2.9%), Tactical Military Intelligence Summaries (≈0.9%), as well as other types of Imagery Interpretation Reports.

6.2 HPS Software Development

This section presents an overview of the HPS framework, the major relevant HPS software components, and our accomplishments with regard to enhancing and further developing the HPS software for the IFS application domain and integration into the IFS.

6.2.1 Information Extraction Pipeline Framework

The purpose of the information extraction pipeline framework is to control the processing of the information extraction components. The diagram below illustrates the progression of the pipeline, and the following paragraph briefly describes the components in the pipeline.

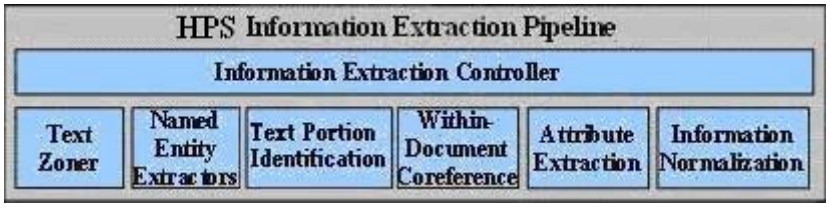

Figure 7 HPS Information Extraction Pipeline

The Text Zoner feeds the usable portion of each document to the Named Entity Extractors, which consist of the Pattern-Based Named Entity Extractor and the BBN IdentiFinder. The Text Portion Identifier then breaks the document up into segments of structured and prose text. The Within-Document Coreference component resolves and links entities that appear multiple times in a document. The Attribute Extraction module assigns characteristics to the identified entities, and the Relationship Extraction module determines relationships and interactions among the entities. Finally, the Information Normalization module normalizes the entity names and the attributes into standardized forms.

More information on each of the individual components is contained in the sections that follow.

6.2.2 Text Zoner

As part of the HPS information extraction processing, it is essential for the HPS to be able to automatically recognize the parts of a document/message and certain text segments. These include:

- Tagged information fields, such as subject, source, distribution list, date, country, etc. (tagged with a label such as "SUBJ:", "SOURCE:", etc.);
- The free text prose portions;
- Security classification labeling;
- Structured parts such as tables, lists, etc.;
- Separators, footnotes, etc.; and
- Text that is extraneous to the actual content of a document, such as page breaks, page headers, page footers, etc.

Without the ability to recognize these different parts or text segments within a document/message, problems arise. These problems include:

- Missed information,
- Extraneous "noise" data being entered into the database, thereby "corrupting" the database,
- Inaccurate processing by downstream information extraction components, and
- Distracting, counterproductive "noise" data being retrieved from the database and presented to end users during their use of the HPS analysis and visualization tools.

The Text Zoner from Cymfony, Inc. (www.cymfony.com), was adapted by GDAIS to address the issues described above. Requirements fulfilled by the Text Zoner include that it is:

- Capable of processing all the common document types processed by HPS.
- Capable of recognizing and labeling (marking up) all the text segments that are important to downstream processing steps such as the information extraction components, some of which use natural language processing (NLP) technology.
- Modifiable, reliable, and extendible to new document types.
- Extendible to recognize and label new types of text segments within a document.

Key features of the Text Zoner are listed below. For more information, consult the Text Zoner User Manual.

- The Text Zoner is **rule-based** to provide extensibility and modifiability, eliminate any "hard coding" of functionality into the tool, and enable the user-developer to focus on a high-level view of any text zoning task, instead of working with the source code for a program.
- The Text Zoner provides and uses a **Rule Specification Language** for pattern-action rules that includes the full suite of regular expressions for conditional pattern definitions.
- The Text Zoner also includes a **procedure-based capability**, augmenting the rule-based capability, to handle the more difficult text phenomena that are too complex to be described via the regular expressions used for the rule specification capabilities mentioned above.

- The Text Zoner implementation is based on the concept of a finite state transducer (FST) to provide **speed** and **robustness**.

As part of the HPS program, GDAIS developed a rule set for use with the Text Zoner. This rule set was designed to do the following:

- First, identify the type of the document being analyzed.
- Second, identify and mark up the sections of the document.

Document Type Identification. The determination of document type is made based on text strings in the document that are unique to the specified types handled by HPS. If a document type cannot be determined based on the text in that document, the document type 'Default' is assigned to the document. The following document types are identified by the Text Zoner for HPS:

- CIA information reports and documents
- DoD information reports and documents
- USMTF documents, including IIRs
- HUMINT documents with a 'TEXT:' delimiter
- HUMINT documents without a 'TEXT:' delimiter
- Other documents (default)

Document Text Segments Identification. A rule set was developed for each document type to accurately determine the boundaries of the text segments expected within the respective document type. The text segments located within each document type include the Header, Footer, Security Section, Document Text Section, and Prose Text Section. Rules were also developed to locate and remove text that should be ignored by downstream components (e.g., the natural language processing component) such as page breaks, page headers/footers, and within-document security classification markings.

6.2.3 Text Portion Identifier

Based on an analysis of the example set of documents provided for the HPS Program, it was apparent that there is a need to process tables and other types of structured text that appear in the documents. In order to do this, it was necessary to differentiate between what is prose and what is structured text within the documents. The Text Portion Identifier (TPI) was designed and implemented to perform this task.

At a high level, when the TPI processes a document, it examines each line to determine whether it is of type prose (part or all of one or more sentences) or structured text (part of a table). For each line, the TPI determines and assigns three confidence measures when determining if a text line is a prose sentence segment or a type of structured text:

1. Context type confidence: A map (the ContextualTPIConfidenceMap) is consulted when determining if the possible major category (sentence, structured, blank line or no line) is

likely given the previous and next line's major category. For example, the sequence of classifications SENTENCE, SENTENCE, SENTENCE is more likely than SENTENCE, STRUCTURED, SENTENCE. The confidence measure values used are High (H), Medium (M), Low (L), and Unknown (U).

2. Context sequence confidence: This confidence measure involves the "minor" classification of a line type as the beginning, continuation, or ending of a major classification (i.e., sentence, generic_structured, and specific structured classifications).

   There are two aspects to this confidence measure: whether the sequence is possible, and if it is possible, its likelihood.

   If a certain classification begins or is a continuation, then the next line must be a continuation or an ending. Possible classifications that do not meet this restriction are filtered out.

   Currently, the likelihood of one sequence (e.g., BEGIN, CONTINUE, END) over another (e.g., END, END, END – where three sentences begin and end on the same line) is not being determined. Data analysis may in the future show that these likelihoods could be useful.

   The confidence measure values used are High (H), Medium (M), Low (L), and Unknown (U).

3. Feature confidence: This confidence measure is based on how good the features used to determine the possible classification are with respect to other features. For example, the feature of having a verb in a line is a better indication of a sentence than the feature of the line having a number of function words (e.g., "of", "as", etc.).

   The confidence measure values used are High (H), Medium (M), Low (L), and Unknown (U). The value is determined based on the feature vector maps used to determine possible line types.

All three types of confidence measures are determined for a possible line type classification. For each major classification, a compilation of confidences is determined which indicates the contributions of each confidence type. For example, for a SENTENCE: (H, U, M), or for STRUCTURED: (L, U, M). The highest feature confidence of a major classification is used in this compilation. The other types of confidence are currently uniform since the sequence confidence is currently used just as a filter (a "U" for Unknown but acceptable classification), and the type confidence is constant across the major classifications.

In the future, this compilation may be used as the basis of a lookup in a map which indicates which major classification should be made according to the relative confidences. Currently, instead of making that decision using a map, the confidences are converted into numeric values (H=3, M=2, L=1, U=0) and summed. The major classification with the highest score is then chosen. The minor classification is then chosen, based on feature confidence.

Currently, the Text Portion Identifier identifies the following types of structured text:

- Order of Battle Tables
- Target Reference Tables

- Imagery Data Tables
- Datelines
- Source Descriptions

If a piece of structured text is not categorized as one of the above types, it is classified as "generic" structured text, which is processed in a more general fashion than any of the more specific types.

6.2.4 Pattern-Based Information Extraction

The pattern-based information extraction component of HPS identifies and extracts those named entities and attributes that can be identified by regular expression patterns. This component is implemented using the underlying engine of the Text Zoner, described above.

As part of the HPS program, GDAIS developed a set of regular expression rules for the Pattern-Based Extractor for application to the document types identified by the Text Zoner (described in above section). The rule set primarily focuses on identification of certain types of dates/times and locations in text. These are the dates/times and locations which follow a pattern that can be expressed using a regular expression. The following types of information are identified by the Pattern-Based Extractor:

- BE Numbers
- Latitude / Longitude Markings
- UTM (MGRS) Numbers
- Dates
- Times
- Date Time Groups
- Phone Numbers
- Country Codes

6.2.5 Abbreviation Handling

Based on data analysis, it was determined that many of the sample JEFX documents use a shorthand for artifact and facility names that was not yet handled by the HPS. As a result, the following abbreviations were added to the pattern-based semantic classification component identifying them as artifacts and facilities, and have also been added to the name normalization component to map them to their non-abbreviated forms:

- RDR: Radar (Artifact)
- LNCHR: Launcher (Artifact)
- VEH: Vehicle (Artifact)
- TRK: Truck (Artifact)
- MSL: Missile (Artifact)
- BLDG: Building (Facility)
- BNKR: Bunker (Facility)

- FAC: Facility (Facility)

6.2.6 NLP Information Extraction

The following subsections describe the information extraction technical approaches that are used in and applied by the Natural Language Processing (NLP) component.

6.2.6.1 NLP Prose Entity Extraction

A Lexicon Lookup module represents one of the approaches used by the NLP component to identify and extract entities. The lexicon lookup mechanism takes every syntactic group identified as a noun group by the Ramshaw Noun Grouper component of HPS, and searches for the noun group terms in the lexicon. The HPS lexicon is implemented as a set of tables within the HPS Database. The lexicon contains sets of names for people, organizations, locations, and military equipments. The lexicon can be expanded using the HPS Domain Porting Tools or by manually adding to terms to the lexicon tables in the HPS Database.

Semantic class inference using pattern-language processing is another portion of the NLP module that is used for entity extraction. The semantic class inference module is based on the idea that there are certain keywords that can give clues as to the classification for a group. For example, in the HPS document set, if the word 'tank' appears, it is a relatively safe assumption that the noun group in which the word 'tank' appears is a piece of military equipment. For example, even if the term 'ABC123 Battle Tank' was missed by IdentiFinder and didn't appear in the term lexicon, it would still be caught as a named entity using semantic class inference on the word 'tank'. The current set of semantic class inference rules catch entities of type Person, Organization, Artifact (military equipment), Facility, and Money.

6.2.6.2 NLP Prose Attribute Extraction

6.2.6.2.1 Frame-Based Extraction

A Frame-based approach is the primary approach used by HPS when processing prose text portions of documents to perform attribute extraction and assignment. Frames are used to extract attributes based on their context and semantic class, as long as they are in the same sentence. The attributes that are currently extracted from prose text are quantity, time, date, and location. The algorithm, which was originally developed for another effort, uses frames and was enhanced to find and associate attributes as part of the HPS program. This work is based on and uses the resources of the Berkeley FrameNet Project, http://www.icsi.berkeley.edu/~framenet.

A frame is a data structure that corresponds to an event, relationship, or attribution. The frame consists of a target, such as "meeting", "president", or "artifact", and a set of frame elements, such as the attendees of the meeting, the name of the president, or the location of the artifact. We acquired a set of frames, target words, and frame elements from Berkeley's FrameNet Project. We then added software whose purpose is to automatically fill the frame elements, that is, to find expressions in the text that correspond to the elements of the frames. To accomplish this, we

provided each frame element with a set of *search strategies.* For example, to find the location of the artifact, a strategy could be to search for appropriate prepositional phrases within the text segment (e.g., “in/near Baghdad”).

Each frame element has its own set of search strategies, ranked by their reliability, as well as a selectional *preference* and a selectional *restriction*. Each search strategy returns a set of possible words, ranked in order of confidence. The selectional *restrictions* filter the outputs of the search strategies, removing words of the wrong semantic or syntactic type. The outputs of the search strategies are combined and stored with the element. The selectional *preference* is then used to choose the best candidate from among the possible words, again based on semantic and syntactic types. In this way the confidence derived from the search strategy is combined with the confidence derived from the selectional preference.

Frames are flexible in that they are triggered by certain keywords and semantic classes that indicate a relationship of some type to another entity in the sentence. Frames are not constrained by precise word sequences, although they can take sequence information into account. As such, frames can be used to identify a wide range of prose phrases that would indicate attributions.

Frames were used in HPS to identify the following types of attributes:

- Quantity
- Location
- Date
- Time
- Confidence
- Facility (if an entity was determined to be inside a building, the name/description of that building would be the ‘Facility’ attribute)
- Color
- Size

To achieve this, each entity type (Artifact, Organization, Facility, and Person) was assigned its own frame. The frame was triggered by semantic class identification—that is, if any entity in a sentence was found to be of type Artifact, the ‘Artifact’ frame was triggered, if an Organization entity was found, the ‘Organization’ frame was triggered, and so on. Search strategies for the different attribute types were developed for each frame, and code was written such that if such a search strategy was successful, the appropriate attribute would be identified and extracted.

6.2.6.2.2 Pattern-Based Extraction

Pattern-action rule-based approach is also used by HPS to find attributes in prose text and assign them to corresponding entities. For example, if the word before a piece of equipment is a word used to describe size, a ‘size’ attribute is assigned to that piece of equipment with a value expressed by the ‘size’ term. As part of the HPS program, patterns were developed to identify and extract attributes of type size, quantity, color, and location and assign them to corresponding entities of type equipment.

6.2.6.3 NLP Structured Text Entity and Attribute Extraction

For the predefined 'known' types of structured text recognized by the TPI (i.e., Order of Battle tables, Target Reference tables, Imagery Data tables, Datelines, and Source descriptions), each row consists of an entity mention in one column and attributes of the entity displayed in the other columns. The Structured Text Processor-Extractor component, which performs information extraction on structured text, will extract the attributes from the appropriate "cells" in any row and assign them to the piece of equipment or facility mention in the corresponding cell of the same row, based on the columns in which they reside. For example, one of the defined structured text types identified by the TPI is an Order of Battle table in which the first column contains quantities, the second column contains equipment, and there is an optional third column for additional information (often used for location). The following is an example of such a table:

```
4    ABC123 Battle Tanks   (GEO: 234345N 0183423E)
1-5  Stake Bed Trucks      (North of Atlantis)
3    APCs
```

Since this type of table is well-defined, it is known that for any row, column 2 is an artifact, column 1 is a quantity associated with that artifact, and column 3 (if it exists) is additional information related to that artifact. So, in this case, columns 1 and 3 would contain attributes of the entity represented in column 2.

In unknown types of structured text, two methods are applied to try and determine the type of attribute or equipment in a column. First, the text of each structured group is analyzed to try and determine its contents. For example, a five-character group in which the first four characters are digits and the last character is a 'Z' is a time. If the content of the group is unrecognizable, other entries in the same column will be checked, and if the content type of any of those other entries can be determined, then it is assumed that every other entry in the same column is of that type.

6.2.7 Attribute Normalization

Normalization of location, date, and time attributes was developed as part of HPS to provide attribute values in standardized forms (rather than simply using the text that expresses the attributes as it appears in the documents). Such standardized forms are required for downstream components/tools such as database search, timeline visualization tools, and geographic map overlay visualization tools.

Locations that are attributed to an artifact are converted into four separate attributes that are assigned to that artifact: Latitude, Longitude, Latitude_Error and Longitude_Error. The Latitude and Longitude attributes are decimal values of the latitude and longitude of the artifact, in degrees, where negative values mean South for Latitude and West for Longitude. Latitude_Error and Longitude_Error represent the margin of error for the Latitude and Longitude measures respectively, measured in meters. Locations that appear in the documents as UTM / MGRS numbers are converted to latitudes and longitudes using NIMA's GEOTRANS software.

A lookup algorithm was implemented to normalize geopolitical entities (e.g., city, town, administrative division names) to lat/long coordinates. The NGA GeoNET Names Server is the source of the location lists, available at http://earth-info.nima.mil/gns/html/ .

Since using one master list of locations would be too costly in terms of memory and speed to implement, we separated the location lists by country. When a lookup is needed, an algorithm runs to decide, using context, which country's location list should be loaded. That country's information then stays resident in memory until it is determined that a different country should be loaded. The lookup then occurs based on the name of the geopolitical entity (using the GeoNET Server's "FULL_NAME_ND" field which omits non-ASCII punctuation), a check is performed to ensure that a found entity in the location list meets the additional search criteria, and then the latitude and longitude attributes are created and assigned as per the previous normalization techniques.

Dates and Times that are attributed to artifacts are converted into three attributes which express the combined date and time at which the entity was observed: Date_Time, Time_Before_Error and Time_After_Error. Date_Time is an expression of the combined date and time of the sighting, in the format *YYYYMMDD***T***hhmmss***Z**. Time_Before_Error and Time_After_Error are numeric values representing an error window expressed in terms of a number of seconds.

6.2.8 New Text Viewer Executable

A new standalone version of the HPS Document Text Viewer was created for integration with the WebTAS visualization toolset for JEFX-04. The HPS Document Text Viewer displays a selectable (mouse-sensitive) list of all the entities (e.g., artifacts, facilities, organizations, people) in the upper-right hand corner of the Text Viewer window. When an entity is selected in this selectable list, the occurrences of this entity are highlighted in the document text, which is displayed in the lower half of the Viewer window. All of the attributes related to the selected entity are also highlighted in the text of the document, using the same color as the entity highlight.

6.2.9 HPS-GIP Integration

In collaboration with Northrop Grumman, an approach to integrating HPS and the Generic Intelligence Processor (GIP) so as to use the GIP to extract entities and attributes from strictly formatted text (specifically, USMTF messages) was designed and implemented. This integration was accomplished based on JEFX-04 requirements. The implementation works as follows:

- The document is loaded into the HPS Document Loader.
- The pre-existing contents of the GIP output directory are deleted.
- The loaded document is copied to the GIP input directory.
    - The GIP input directory is a folder that is shared by the UNIX system running the GIP and the Windows system running the HPS.
- The GIP processes the document based on its type, determined by the MSGID field.
    - If no MSGID field exists, then no GIP output is generated.

- The GIP writes XML files to the GIP output directory.
  - The GIP output directory is also a shared UNIX / Windows directory.
- HPS, in addition to the processing it previously performed, creates time, date, and location attributions for any equipment or facilities identified by the GIP.

The GIP-based extension of HPS currently processes strictly formatted text in IIR (Image Report) and MISREP (Mission Report) messages. Future plans include further extending the GIP and HPS to process RECCEXP, SENSOREP, and TACELINT messages.

6.2.10 Integration with the IFS

One of the most important requirements in the development of HPS was that it must be able to be integrated with other components as part of the IFS. The figure below illustrates the TUT IFS and shows the HPS (called IEPS in the figure) as a component.

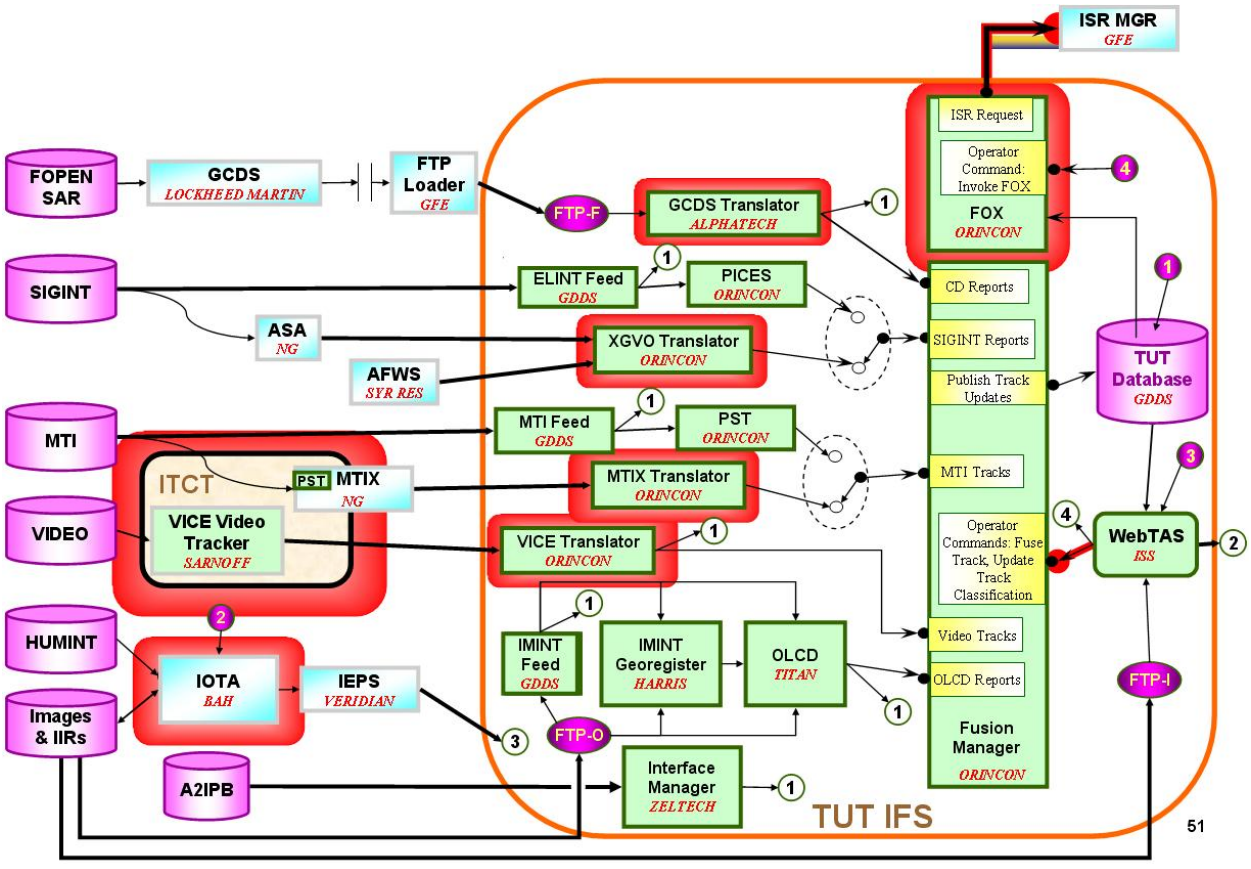

Figure 8 TUT IFS-HPS Integration Overview

The processing stream involving HPS is as follows:

1. HUMINT (HUMan INTelligence) and IMINT (IMage INTelligence) messages are fed to the IOTA (Infrastructure Operational Tool Access) system, developed by Booz-Allen-Hamilton and Northrop Grumman, via e-mail.
2. The IOTA system places documents to be processed by HPS into a specific directory.
3. HPS pulls the documents from that directory and processes them, performing information extraction including entity extraction and the normalizing and assigning of attributes to the extracted entities.
4. HPS writes the results to the HPS Database, from which the IFS Fusion Manager, developed by Lockheed Martin – Orincon, pulls the information.
5. IFS fuses the data from HPS with other types of intelligence that have also been processed, for a complete portrayal of all known intelligence.
6. IFS sends this fused information to WebTAS, where it is displayed in map form, and from which the source documents for any HUMINT or IMINT can be viewed by the human operator.

As part of the integration effort, several tasks were performed. A directory monitor was built as part of the HPS package, so that when the IOTA system places messages into the directory, they can be automatically retrieved and processed by the HPS system. The directory monitor polls the folder once every second, and if a file is found in the folder, it is added to the HPS queue and moved to a different folder for processing. These are customizable via a configuration file.

Normalization of attributes and equipment names was also necessary so that the information gleaned from the documents by HPS could be fused with other intelligence. Normalization is described briefly in Section 6.2.7 of this document. Finally, a streamlined, standalone version of the HPS Document Viewer was created so that documents could be viewed by the IFS operator when selected in WebTAS. This Document Viewer enables the operator to view the contents of any original document/message and displays the identified entities and assigned attributes in the document.

7 Statistical-Based Information Extraction

As part of the HPS effort, BBN evaluated its train-by-example named entity extractor, IdentiFinder, on a training corpus of 1,422 Government-furnished messages. The evaluated system was delivered for integration into the HPS operational prototype. An alternative entity detection method, descriptor classification, was applied to and evaluated on the same message corpus. A third, experimental technology, currently being developed at BBN was considered for target extraction, but on a different, larger corpus.

In the following subsections, the message set and entities to be extracted are discussed. The next three subsections present each of the three entity detection methods. The final subsection compares the performance for each of the approaches.

7.1 Messages to be Processed

The original expectation was that the contractors would receive no fewer than 2 million words of running prose documents that are broadly representative of the texts to be targeted for information extraction. At least 50% of the documents were to be unclassified. The remaining documents were not to be classified above the SECRET (S) collateral level.

During the first phase of the effort, it was determined that unclassified documents did not contain the surface target information that the program intended to extract. Consequently, all work was performed with S documents. Ultimately, a training corpus of 1,422 unique S documents containing approximately 1.2 million words of prose text was delivered to BBN for experimentation and development.

Further discussions explored the relevancy and need to process COMINT messages, messages controlled in special compartments, and kleiglites. For logistical and cost reasons, processing these documents was deferred to possible follow-on efforts.

7.2 Information to be Extracted

Fourteen entity types were targeted for information extraction under the HPS effort. They are listed in the table below. Initially, the equipment category was subdivided into Fixed Equipment, Mobile Equipment, and Transportable Equipment. Based on discussions with collaborating analysts, one an active analyst on the staff of NASIC and another a retired analyst on the staff of Booz Allen Hamilton (BAH), it was quickly determined that it is difficult for a human, and beyond the scope of automatic means such as BBN's IdentiFinder, to differentiate fixed, mobile and transportable equipment. Consequently, the three subcategories were collapsed into a single equipment category.

Table 6 Extracted Entity Types

Entity	Description
BE Number	BE Numbers in text
Date	A date with granularity greater than or equal to 24 hours
DTG	A conjoined time and data.
Equipment	Tangible equipment or materiel
Facility	Facilities such as A.F. Bases
Facility Description	Noun phrase descriptions of facilities, .e.g. “the air force base”
Geo-Coordinate	Geo-Coordinates or lat-longs
Geo-Political Entity	A location with a Government and population associated with it
Location	Geographic locations
Organization	Proper names of Organizations
Organization Description	Noun phrase descriptions of organizations
Person	Proper names of people
Person Description	Noun phrase descriptions of people
Time	A time with granularity less than 24 hours

Manual extraction (annotation) was performed on all provided messages. The table below lists the occurrence frequency of each target entity type within the training corpus. Generally, the greater the number of training examples, the better the expected performance; however, poor performance extracting one type can adversely affect extraction of other types.

Table 7 Entity Occurrence Frequency in a 1.2 Million Word Training Corpus

Entity Type	Frequency
BE Number	2,633
Date	13,082
DTG	571
Equipment	41,383
Facility	11,539
Facility Description	31,069
Geo-Coordinate	4,809
Geo-Political Entity (GPE)	26,330
Location	2,835
Organization	18,523
Organization Description	12,652
Person	5,771
Person Description	16,025
Time	4,492
TOTAL	191,714

7.3 IdentiFinder

IdentiFinder is a train-by-example, HMM-based named entity extraction software system. Demarcated examples of entity mentions are provided to IdentiFinder as they occur in text as training data. The IdentiFinder trainer statistically infers the relevance and weights of lexical, morphological, and contextual word features that support a word sequence being classified into a semantic category. The weights are instantiated into an IdentiFinder model. The model is applied to new messages, extracting those word sequences most likely to be one of the entity types for which the system was trained.

7.3.1 Annotation and Training

Training and annotation were begun at the AFRL Rome Research Site. BBN delivered hardware, annotation software, and quality assurance tools to the AFRL I3F facility. Three local persons, two BAH contractors and one AFRL staff member were trained to annotate the initial 526 messages. BBN traveled to AFRL on several occasions to train and lead the annotation effort.

In parallel with the effort, BBN annotators began annotating the same 526 message corpus at its Cambridge facility. Initially, overlapping document sets were annotated to allow for inter-rater reliability and other quality assurance measures to be performed.

AFRL-based annotation and Quality Assurance (QA) was performed on 526 messages. BBN took receipt of the 526 messages, performed further QA tests on them and used them to bootstrap annotation for an additional 900 documents. The final annotated corpus consisted of 1,422 unique documents comprised of 1.2 Million words.

An experiment was conducted during the second phase of the effort to assess the performance affects of training IdentiFinder with large quantities of domain-generic training materials. Over four million words of Wall Street Journal and other newswire articles previously annotated with person and organization demarcations were used to augment IdentiFinder training. It was hoped that the large infusion of training would increase the model's vocabulary, thereby increasing performance without diluting the positive effectives of the domain specific training. This experimented ended by showing that adding generic training to 1.2 million words of domain specific training does not improve performance.

7.3.2 Tools

Initially, BBN delivered its IdentiTagger™ annotation tool to support mouse-based document annotation. One shortcoming of this tool is that it relies on color. One of the three AFRL-based annotators and an AFRL reviewer were color blind. While other users found the IdentiTagger to be easy and intuitive, these staff members could not use the tool.

In support of the AFRL annotation effort, BBN developed and delivered software support tools that (1) automated bootstrapping annotation from existing models; (2) performed heuristically driven QA; and, (3) automated QA testing. Automatic QA was affected by training an

IdentiFinder model on an annotated particular corpus, stripping the annotations, applying IdentiFinder to the stripped corpus, and compare IdentiFinder output with the original, manual annotation using NIST's MUC Scorer. Each of the tools was delivered to the AFRL I3F facility along with documentation and training.

During the course of the HPS Program, BBN released version 3 of its IdentiFinder software. This version was delivered along with a user manual to both GDAIS and AFRL.

7.4 Syntactic Based Approaches

IdentiFinder's HMM uses bigrams (neighboring word context) to determine entity boundaries. An alternative approach is to syntactically parse each sentence and use syntactic cues, such as the extent of noun phrases, to determine the extent of entities. Syntactic parse information also allows one to identify and exploit syntactic neighbors in addition to lexical ones. For example, in the fragment, "Michael, a BBN employee, stated…," "Michael" is syntactically adjacent to "stated" event though they are lexically separated by five tokens.

Syntactic parse methods are most useful for identifying descriptive entity references such as "the former dictator," as opposed to names or pronouns. They are also necessary for resolving co-reference among entity mentions; however, that capability was out of the scope of this effort.

To perform syntactic parsing and syntactic based entity extraction, also referred to as descriptor extraction, BBN employed components of its SERIF™ (Statistical Entity and Relationship Information Finder) Natural Language Processing (NLP) toolkit.

SERIF proper name extraction uses an embedded version of IdentiFinder. SERIF syntactic parser models are trained from LDC (Linguistic Data Consortium) syntactic annotation of Wall Street Journal articles. In addition, training sentences exemplifying areas in which the parser had difficulties with HPS data were synthetically constructed, annotated, and used to augment the parser models.

7.5 Automatic Cluster-based Approaches

During the third phase of the effort, experiments were performed applying nascent training and decoding algorithms developed under other efforts that require minimal human resources. The algorithms synthesize elements of mutual-information based clustering, voted perceptron discriminative training, and active learning. They will be described in a forthcoming HLT paper. The training process is comprised of two phases. During the first phase, fully automatic word clustering algorithms are applied to a large corpus – at least 100 million words – of messages. The result of this phase is a binary tree of words, with similar words closer together in the tree.

During the second phase, a user is prompted first for examples of the entity to be extracted. Using a combination of the user seeded examples and cluster-based features from the first phase, the user is iteratively prompted to annotate sentences for which the computer models are least decisive. At each iteration, the extraction models are updated.

The aforementioned training process stands in stark contrast to IdentiFinder, which requires manual annotation of at least 500K words of message traffic to train a system. With guideline development, staff training, and quality assurance, this requires significant resources. For the HPS program where initial effort had a particularly low inter-rater reliability rate, the resource requirements were especially onerous.

BBN conducted two clustering experiments. Due to the data requirements of at least 100 million words, the experiments were performed with FBIS and AP Newswire messages. The following list is an excerpt from the clustering output when run against 50 million words of FBIS documents.

Table 8 Fully Automatic Clustering Output

…
helicopter
warplane
chopper
Su-24
Tavor
hand-grenade
car
pipe
tube
cylinder
projectile
shoe
trailer
truck
Shaheen-I
lorry
minibus
rickshaw
…

Notable about these results is that with no human intervention, the system determined that helicopter, warplane, chopper, and SU-24 are all closely related. This is especially significant given that FBIS documents, unlike those used in the HPS effort, do not typically discuss military equipment. BBN expects that performance will improve in proportion to mention frequency.

Further analyses must be performed to determine whether a vehicle or equipment extractor can be built on these results. Based on initial inspection of the clustering algorithm's output, BBN is optimistic that unsupervised structure will prove useful for reducing dependence on annotated data for information extraction tasks and entity extraction in particular.

7.6 Extraction Performance

Extraction performance was measured for each of the aforementioned technical approaches: IdentiFinder, IdentiFinder augmented with generic training, and syntactic-parser-based. Cluster-based approaches are still too immature for meaningful, quantitative evaluation. Performance is given in the tables below.

Table 9 Extraction Performance (F-Score)

	IdF	IdF: HPS data augmented with ~4M words WSJ and other newswire PER & ORG	Syntactic Parse PER, ORG & FAC
Test-on-Train	95.4	~95	91.5
Fair-Test	84.8	~84	79.6

Evaluating performance on training messages (test-on-train) establishes upper bounds on performance. It can also be used to indicate annotation inconsistencies or annotation guideline ambiguities. For this effort, 95.4 and 91.5 F scores indicate that adequate annotation consistency and guideline clarity were achieved. On Wall Street Journal (WSJ) messages – an industry baseline – test-on-train scores are roughly 98 F.

An IdentiFinder score of 84.8 is below that of the low 90's that is achieved extracting proper names from Wall Street Journal articles. Given the extended breadth of the source materials, the semantic types, and the syntactic types (common noun phrases in addition to proper names), these results are well within, if not exceeding expectations formed at the project outset.

Augmenting IdentiFinder training with generic markup for persons and organizations did not affect extraction performance. In fact, in early experiments, it had a negative impact on performance. This result reinforces our belief that representative training is critical for extraction performance.

Syntactic parse based methods performed 4-5 points worse than IdentiFinder. Several theories were considered to explain this. Ultimately, the conclusion was reached that both proper name references and descriptive references can be determined using the local (bigram) modeling that IdentiFinder performs well. The additional information provided by syntactic parsing was not useful to the entity identification task.

As discussed earlier, cluster-based methods are in their infancy. As these algorithms continue to mature BBN expects to better understand their strengths and to what problems they are best suited.

8 Known Problems and Open Issues

This section summarizes the known problems and open issues for the HPS system and program.

8.1 Location Normalization

Basic Encyclopedia (BE) Numbers are currently extracted from text documents/messages and assigned to entities as attributes of type 'Location'. Since BE Numbers are not actually measures of location, there is no numeric conversion that can be performed to convert them to the normalized latitude and longitude attributes. An interface with a resource such as the MIDB is necessary to map the BE Numbers to locations. The MIDB contains a catalogue of all known worldwide facilities, their BE Numbers, and their corresponding locations.

Geopolitical entities (e.g., town names, geographic feature names) are also extracted from text documents/messages and assigned to entities as attributes of type 'Location'. Again, there is no way to mathematically convert a town name to a latitude and longitude, so a lookup mechanism is required for location normalization. A preliminary lookup mechanism has been implemented to perform this mapping, but in its current incarnation, telling the difference between Paris, France and Paris, Texas is not possible—the name 'Paris' would most likely be normalized to whichever 'Paris' is found first in the database. Further design and development of this module is required to provide more accurate results for the conversion of geopolitical entity names to latitude / longitude representations.

8.2 Negative Inference

There are cases in which entities are mentioned in the text of a document/message, but the reason that they are mentioned is to note that they are no longer at a certain location. For example, a sentence in a document could say:

```
The ABC123 Battle Tank can no longer be seen at the DEF456 Repair Facility at
123456N 0123456E.
```

The current version of the HPS system will extract the entity 'ABC123 Battle Tank' and assign it a location of '123456N 0123456E', even though the text indicates that the tank seems to no longer be at that location. The HPS is not able to correctly handle the negative verb phrase. Additional work is required to address this issue.

8.3 Interaction between Prose and Structured Text

There is currently some assignment of attributes identified in structured text to entities found in prose text, particularly header information that is outside the scope of the "Free Text" prose section of a given document. Inside the "Free Text" section of many documents, however, there are pieces of structured text that function as "sub-headers"—that is, header information for a small piece of text that follows it. For example, many Target Reference Tables serve as sub-

headers, as they often give the name and BE Number of a facility, followed by a brief paragraph detailing pertinent information about that facility. An example:

```
ABC123 Repair Facility                    BE 1234-98765

There were 10 ABC123 tanks and 4 stake bed trucks at this facility on 30
February, 2004.
```

In the above piece of text, the BE Number would be attributed to the facility, and the quantity and date attributes would be attributed to the tanks and the trucks, but the BE Number would not be attributed to the tanks and trucks. Similarly, the prose immediately preceding a piece of structured text might contain information about entities in that structured text. For example:

```
The following equipment is located at 123456N0123456E:

1 – Stake Bed Truck
8 – ABC123 Battle Tanks
2-3 – XYZ987 Battle Tanks
```

The link between the location expressed in the prose text and the equipment in the structured text would be missed by the current version of the HPS system.

8.4 Equipment Co-Reference

When a person refers to a piece of military equipment using a phrase such as ‘an ABC123 Battle Tank’, the proper name in the phrase is not unique to a particular instance of a tank. Instead it is the name of the class or type of tank. The phrase ‘an ABC123 Battle Tank’ does not uniquely identify the entity to which it refers. Rather, it declares the presence of an instance of the ‘ABC123 Battle Tank’ type. This means that two or more mentions that include ‘ABC123 Battle Tank’ may or may not refer to the same tank in a document. The current HPS system does not handle coreference for these types of mentions. Since equipments are commonly expressed using such phrases, however, as a result, equipment coreference is virtually nonexistent in the current HPS.

There are many cases in the sample data for the HPS program where different attributes of the same equipment are explained in different sentences. The coreference capability of HPS needs to be improved so as to handle these types of cases and enable the HPS to exploit the type of information reported in the sample documents/messages. Enhancement of equipment coreference would help improve attribution of equipment by combining some mentions of military equipment into one entity. For example, in the following passage:

```
Three ABC123 Battle Tanks were seen at 123456N0123456E. These tanks were
spotted on 8 December, 1993.
```

In the current HPS system, processing this passage would output two entities: ‘ABC123 Battle Tank’ would have a quantity attribute of ‘three’ and a location attribute of ‘123456N0123456E’, and ‘tank’ would have a date attribute of ‘8 December, 1993’, but there would be no indication that the two identified entities referred to the same tank.

8.5 Database Streamlining

The design of the HPS Database is good in that it provides the flexibility to accommodate any types of attributes and associate them with any entities. However, the current design of the HPS Database is complex in this regard and includes a fairly large number of linked tables that contain the information for attributions. The Attribute table contains the attribute itself, the Attribution_Instance table contains the entity that the attribute is being attributed to, the Attribution Table contains the information to link the two previous tables, and then the Meta_Information table contains document information and the Text_Reference and Expression_Link tables contain offset information.

When processing a large number of documents, we learned that the HPS Database attribute tables can become highly populated and cause problems. The database query that is used to access the attribute information is fairly complex, and after a number of documents have been run, the query takes so long as to time out. The current database design simply does not accommodate large amounts of attribution data.

At JEFX-2004, a solution to this problem was implemented, and a new table was added to the database. This table summarized all of the attribute information for the entities found in the document, thus greatly reducing the number of tables that need to be accessed to compile all of the information in the tables mentioned in the above paragraph. This is a nice short-term solution, but this issue should be examined further. A better long-term solution might include redesign of the tables that hold attribution information, as the current solution is faster, but redundant when combined with those other tables.

8.6 Throughput

Processing large volumes of documents through HPS revealed an important issue: The speed of the HPS system is far slower than what might be necessary for a real-time product in the field. With the introduction of the GIP as an integral part of the HPS, a typical document, which previously took 2-3 seconds to process, can now require thirty seconds or more to process. The lack of speed is the result of a combination of a number of factors: The speed of the GIP itself, the need for a more efficient interface between the main HPS controller and the GIP, the need for more speed optimization in the NLP component of HPS, and the need for a filtering mechanism to filter out duplicate messages or messages for which GIP processing would produce no useful results. Addressing these needs would greatly improve overall throughput performance.

8.7 Statistical-Based Information Extraction

All of the experiments performed by BBN and reported in the previous section were performed in a theoretical context. No end-to-end system test or user feedback was provided. This raises issues of how well the theoretical performance measures track user requirements. Any further work would greatly benefit from review and feedback by users. Furthermore, it would enable the

system developers to emphasize those aspects of the system most important to the users, thereby resulting in a more valuable product.

The clustering experiments were performed with a data set fundamentally different than those used by the rest of the system. This was because of the unavailability of adequate quantities of appropriate message traffic. Given the promise of clustering technologies, a critical open issue is to determine how well clustering methods perform when applied to HPS relevant message traffic that contains correspondingly comprehensive references to targets.

9 Lessons Learned

This section presents the more significant lessons learned during the HPS program and development process.

9.1 Information Sources

Before the sample document corpus from the Government was carefully analyzed, it was expected that the majority of the relevant information would need to be extracted from the prose text found in the documents. However, we learned that although prose text is indeed the most prevalent of the text types in many of the document types, there is much more structured text in the documents than was originally expected, and the structured text, where it occurred, turned out to be much richer in pertinent information than the prose text. By structured text, we mean the tables, lists, etc., that occur within prose text (not USMTF fields). As such, the lesson learned was that structured text could not be ignored. As a result, software capabilities were developed to process and extract information from structured text and these capabilities were integrated into the HPS. These capabilities include the Text Portion Identifier and the Structured Text Processor-Extractor, among others.

9.2 Design Modification

The Text Portion Identifier is an integral part of HPS, since many of the other downstream HPS components rely on accurate identification of prose and structured text. The initial version of the TPI was design to process text on a line-by-line basis, with a number of conditional statements that were designed to first decide whether a given line was prose, and then, if the line did not have any of the characteristics of prose text, the TPI would then decide what type of structured text the line would best be classified as. Subsequently, it became obvious that identifying the structured text after identifying the prose is not the most effective approach. As a result, a change in technical approach was made, and the redesign resulted in the current implementation of the TPI, which uses a feature-based design. The new approach uses the features to simultaneously check to determine whether a given line has more features of structured text or prose text. This allows for some types of structured text which contain some of the features of prose text to be identified far more effectively than in the first version of the design. Since the features themselves are contained in text file maps, they can be tweaked without code changes, making them much easier to customize to new document sets as needed.

9.3 Statistical-Based Information Extraction

IdentiFinder is an appropriate tool to extract proper name mentions of persons, organizations, etc., for which it was designed. Performance degrades, but still supports mission-critical information extraction when its scope is expanded to include descriptive references and entities that are generally not named, such as equipment.

IdentiFinder’s reliance on manually annotated training materials continues to be a limitation. Multiple users were not able create training materials without significant BBN adjudication. The clustering methods demonstrated potential for addressing this limitation; however, further work must be performed to determine their applicability and performance.

References

1. General Dynamics AIS, *HUMINT Corpus Evaluation Working Document*, dated 23 December 2002.

2. General Dynamics AIS, *Consolidated Requirements and High Level Design Document (CHLRD) for the HUMINT Processing System*, August 2004.

3. General Dynamics AIS, *Data and Knowledge Resource Document (DKRD) for the HUMINT Processing System*, September 2004.

4. General Dynamics AIS, *Software User Manual for the HUMINT Processing System*, September 2004.

5. BBN Technologies, *BBN IdentiFinder User's Guide*, Release 3.0, September 2003.

6. NGA GeoNet Names Server, http://earth-info.nima.mil/gns/html/

7. Berkeley FrameNet Project, http://www.icsi.berkeley.edu/~framenet

8. Cymfony, Inc., "Text Zoner User Manual," October, 2001.

Appendix - Acronyms and Abbreviations

AC2ISRC	Aerospace Command and Control and Intelligence, Surveillance, Reconnaissance Center
AFRL	Air Force Research Laboratory
AIES	Automated Information Extraction System
AOC	Air Operations Center
AP	Associated Press
ATD	Advanced Technology Development
BAH	Booz-Allen Hamilton
BDA	Bomb Damage Assessment
BE	Basic Encyclopedia
BTG	Basic Target Graphics
CAOC-X	Combined Air Operation Center - Experimental
CD	Change Detection
CDRL	Contract Data Requirements List
COMINT	Communications Intelligence
COTS	Commercial Off-The-Shelf
CRHLD	Consolidated Requirements and High Level Design Document
CSV	Comma Separated Value
DCARS	Document Content Analysis and Retrieval System
DCGS	Distributed Common Ground Station
DOD	Department of Defense
DTG	Date/Time Group
FBIS	Foreign Broadcast Information Service
FOPEN	Foliage-Penetrating (Radar)
FST	Finite State Transducer
GDAIS	General Dynamics Advanced Information Systems
GIP	Generic Intelligence Processor
GMTI	Ground Moving Target Indicator
GOTS	Government Off-The-Shelf
GPE	Geo-Political Entity
GRAPHREP	USMTF Graphic Representation Message
GUI	Graphical User Interface
HMM	Hidden Markov Model
HPS	HUMINT Processing Subsystem
HTML	Hypertext Markup Language
HUMINT	Human Intelligence
IAA	Intelligence Analyst Associate
ID	Identification
IDBTF	Integrated Database Transaction Format

Appendix - Acronyms and Abbreviations (Continued)

IdF	IdentiFinder
IE	Information Extraction
IFS	Intelligence Fusion System
IIPR	Integrated Intelligence Production Report
IMINT	Imagery Intelligence
INT	Intelligence
IOTA	Infrastructure Operational Tool Access
ISR	Intelligence, Surveillance and Reconnaissance
ITEA	Intermediate Text Exploitation ATD
JEFX	Joint Expeditionary Force Experiment
JWAC	Joint Warfare Analysis Center
LDC	Linguistic Data Consortium
MASINT	Measurement and Signatures Intelligence
MIDB	Modernized Integrated Database
MISREP	Mission Report
MSGID	Message Identification
MTI	Moving Target Indicator
MUC	Message Understanding Conference
NASIC	National Air and Space Intelligence Center
NIMA	National Imagery and Mapping Agency
NIST	National Institute of Standards and Technology
NLP	Natural Language Processing
QA	Quality Assurance
RECCEXP	Reconnaissance Exploitation Report
SAM	Surface-to-Air Missile
SAR	Synthetic Aperture Radar
SDD	Software Design Document
SENSOREP	Sensor Report
SERIF	Statistical Entity and Relationship Information Finder
SIGINT	Signal Intelligence
SRS	Software Requirements Specification
SSS	System Segment Specification
TACELINT	Tactical Elint Report
TEA	Text Exploitation ATD
TIM	Technical Interchange Meeting
TPI	Text Portion Identifier
TUT	Targets Under Trees
USMTF	United States Message Text Format

Appendix - Acronyms and Abbreviations (Continued)

VHF	Very High Frequency
WebTAS	Web-Based Timeline Analysis System
WSJ	Wall Street Journal
XDA	eXtensible Distributed Architecture
XML	Extensible Markup Language